Online Banking & Insurance System Documentation

# 1. Introduction

The Online Banking & Insurance System (FinSecure) is a modern web application designed to provide users with a seamless experience for managing their banking and insurance needs in one secure platform. The system is built using Python, Flask, SQLAlchemy, and features a beautiful, responsive UI with HTML, CSS, and JavaScript.

# 2. Project Purpose

* Enable users to manage multiple bank accounts (checking, savings, credit) and insurance policies (health, auto, home, life) from a single dashboard.
* Provide secure authentication, transaction management, and policy claims.
* Offer a user-friendly, attractive, and responsive interface for both desktop and mobile users.

# 3. Key Features

* Banking:  
  - Create and manage multiple account types  
  - Deposit, withdraw, and transfer funds  
  - View transaction history and account statements  
  - Dashboard with financial statistics and charts
* Insurance:  
  - Create and manage various insurance policies  
  - File and track insurance claims  
  - View policy details, renewals, and upcoming expirations  
  - Dashboard with policy statistics and visualizations
* General:  
  - Secure user authentication (Flask-Login, JWT for API)  
  - Email notifications for important events  
  - User profile management  
  - RESTful API for third-party integrations  
  - Responsive, modern UI/UX

# 4. Technology Stack

* Backend: Python, Flask, Flask-SQLAlchemy, Flask-Migrate
* Frontend: HTML5, CSS3 (custom + Bootstrap 5), JavaScript (ES6+), jQuery
* Database: SQLite (development), PostgreSQL (production-ready)
* Authentication: Flask-Login, JWT
* Email: Flask-Mail
* API: RESTful endpoints with JWT authentication
* Other: Chart.js for data visualization, Flatpickr for date pickers

# 5. System Architecture

Details about the modular app structure, models, forms, templates, and static assets.

# 6. UI/UX Design & Enhancements

Modern responsive layout with Bootstrap 5, accessibility features, custom icons, real-time notifications, and animations.

# 7. How to Use

* Register: Create an account with your details.
* Login: Access your dashboard.
* Banking: Create accounts, deposit/withdraw/transfer funds, view transactions.
* Insurance: Create policies, file claims, view and renew policies.
* Profile: Update your information and change your password.
* Notifications: Stay updated with important events.

# 8. File & Directory Structure Explained

Detailed explanation of files such as run.py, requirements.txt, config.py, and directories like app/, templates/, static/, models/, forms/, routes/, utils/, migrations/, tests/.

# 9. Future Enhancements

* Mobile app version (React Native/Flutter)
* Two-factor authentication (2FA)
* Admin panel for system management
* More payment integrations (Stripe, PayPal)
* AI-based fraud detection
* Multi-language support

# 10. Conclusion

FinSecure is a robust, scalable, and user-friendly platform for managing banking and insurance needs. Its advanced features, secure architecture, and beautiful UI make it suitable for both individuals and businesses.

# 11. How to Run the Project

Instructions for cloning the repo, setting up a virtual environment, installing dependencies, setting environment variables, initializing the database, and running the Flask app.

# 12. Website Content Overview

Overview of pages: Landing, Login/Register, Dashboard, Banking, Insurance, Profile, Notifications.

# 13. How to Use the Website

* Register: Click “Register” and fill in your details.
* Login: Enter your credentials to access your dashboard.
* Banking: Create new accounts, deposit/withdraw/transfer funds, and view transaction history.
* Insurance: Add new policies, file claims, track claim status, and renew policies.
* Profile: Update your personal information and change your password.
* Notifications: Check for important updates and alerts.

# 14. Demo User Credentials

* Demo User:  
  - Username: demouser  
  - Password: Demo@1234
* Admin User:  
  - Username: admin  
  - Password: Admin@1234

# 15. Admin Features

* Access to all user accounts and data.
* Manage and approve insurance claims.
* View system-wide statistics and reports.
* Send notifications to all users.
* Manage banking and insurance products.

# How to Run the Project (Detailed)

1. Clone the Repository:  
- Download or clone the project to your local machine.

2. Set Up a Virtual Environment:  
(Windows)  
```  
python -m venv venv  
venv\Scripts\activate  
```  
(macOS/Linux)  
```  
python3 -m venv venv  
source venv/bin/activate  
```

3. Install Dependencies:  
```  
pip install -r requirements.txt  
```

4. Set Environment Variables:  
- Create a `.env` file or set variables in your shell:  
 - FLASK\_APP=run.py  
 - FLASK\_ENV=development  
 - SECRET\_KEY=your\_secret\_key  
 - (Optional) DATABASE\_URL=sqlite:///site.db

5. Initialize the Database:  
```  
flask db init  
flask db migrate  
flask db upgrade  
```

6. Run the Application:  
```  
flask run  
```  
- Visit http://localhost:5000 in your browser.